


Terms used **synchronous data mirroring storage system**

Found 107,377 of 143,422

Sort results by


[Save results to a Binder](#)

Try an [Advanced Search](#)

Display results


[Search Tips](#)

Try this search in [The ACM Guide](#)
☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)


Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐

1 [The HP AutoRAID hierarchical storage system](#)

J. Wilkes, R. Golding, C. Staelin, T. Sullivan

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles**, Volume 29 Issue 5


Full text available:  [pdf\(1.60 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)


2 [The HP AutoRAID hierarchical storage system](#)

John Wilkes, Richard Golding, Carl Staelin, Tim Sullivan

February 1996 **ACM Transactions on Computer Systems (TOCS)**, Volume 14 Issue 1

Full text available:  [pdf\(1.82 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Configuring redundant disk arrays is a black art. To configure an array properly, a system administrator must understand the details of both the array and the workload it will support. Incorrect understanding of either, or changes in the workload over time, can lead to poor performance. We present a solution to this problem: a two-level storage hierarchy implemented inside a single disk-array controller. In the upper level of this hierarchy, two copies of active data are stored to provide f ...

Keywords: RAID, disk array, storage hierarchy

3 [Improving storage system availability with D-GRAID](#)

Muthian Sivathanu, Vijayan Prabhakaran, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau
May 2005 **ACM Transactions on Storage (TOS)**, Volume 1 Issue 2

Full text available:  [pdf\(700.30 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)



We present the design, implementation, and evaluation of D-GRAID, a gracefully degrading and quickly recovering RAID storage array. D-GRAID ensures that most files within the file system remain available even when an unexpectedly high number of faults occur. D-GRAID achieves high availability through aggressive replication of semantically critical data, and fault-isolated placement of logically related data. D-GRAID also recovers from failures quickly, restoring only live file system data to a h ...

Keywords: Block-based storage, Disk array, RAID, fault isolation, file systems, smart disks

4 [RAID: high-performance, reliable secondary storage](#)

Peter M. Chen, Edward K. Lee, Garth A. Gibson, Randy H. Katz, David A. Patterson

June 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 2

Full text available:  [pdf\(3.60 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index](#)


Disk arrays were proposed in the 1980s as a way to use parallelism between multiple disks to improve aggregate I/O performance. Today they appear in the product lines of most major computer manufacturers. This article gives a comprehensive overview of disk arrays and provides a framework in which to organize current and future work. First, the article introduces disk technology and reviews the driving forces that have popularized disk arrays: performance and reliability. It discusses the tw ...

Keywords: RAID, disk array, parallel I/O, redundancy, storage, striping

5 [LND: a reliable multi-tier storage device in NOW](#)

Yun Mao, Youhui Zhang, Dongsheng Wang, Weimin Zheng

January 2002 **ACM SIGOPS Operating Systems Review**, Volume 36 Issue 1

Full text available:  [pdf\(686.93 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The expensive overhead of synchronous I/O has now become the bottleneck of increasing efficiency of data storage in transaction-based systems. This paper proposes a multi-tier storage device, the LND (Local-Network Ram- Disk) device, which uses idle memories and disks in NOW (Network of Workstations) as persistent repositories so that access latency is transformed from magnetic disk seek-and-transfer latency to the sum of network transfer latency, memory write latency, and asynchronous I/O laten ...

Keywords: high availability, network ram, synchronous I/O, transaction

6 [DISP: Practical, efficient, secure and fault-tolerant distributed data storage](#)

Daniel Ellard, James Megquier

February 2005 **ACM Transactions on Storage (TOS)**, Volume 1 Issue 1

Full text available:  [pdf\(148.11 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

DISP is a practical client-server protocol for the distributed storage of immutable data objects. Unlike most other contemporary protocols, DISP permits applications to make explicit tradeoffs between total storage space, computational overhead, and guarantees of availability, integrity, and privacy on a per-object basis. Applications specify the degree of redundancy with which each item is encoded, what level of integrity checks are computed and stored with each item, and whether items are stor ...

Keywords: Distributed data storage

7 [Interposed request routing for scalable network storage](#)

February 2002 **ACM Transactions on Computer Systems (TOCS)**, Volume 20 Issue 1

Full text available:  [pdf\(363.12 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citings](#), [index terms, review](#)

This paper explores interposed request routing in Slice, a new storage system architecture for high-speed networks incorporating network-attached block storage. Slice interposes a request switching filter---called a μ proxy---along each client's network path to the storage service (e.g., in a network adapter or switch). The μ proxy intercepts request traffic and distributes it across a server ensemble. We propose request routing schemes for I/O and file service traffic, and explore th ...

Keywords: Content switch, file server, network file system, network storage, request redirection, service virtualization


8 [Understanding fault-tolerant distributed systems](#)

Flavin Cristian

February 1991 **Communications of the ACM**, Volume 34 Issue 2

Full text available:  [pdf\(6.17 MB\)](#) Additional Information: [full citation](#), [references](#), [citings](#), [index terms, review](#)

- 9 Designing computer systems with MEMS-based storage
Steven W. Schlosser, John Linwood Griffin, David F. Nagle, Gregory R. Ganger
November 2000 **Proceedings of the ninth international conference on Architectural support for programming languages and operating systems**, Volume 34 , 28 Issue 5 , 5

Full text available:  pdf(439.06 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

For decades the RAM-to-disk memory hierarchy gap has plagued computer architects. An exciting new storage technology based on microelectromechanical systems (MEMS) is poised to fill a large portion of this performance gap, significantly reduce system power consumption, and enable many new applications. This paper explores the system-level implications of integrating MEMS-based storage into the memory hierarchy. Results show that standalone MEMS-based storage reduces I/O stall times by 4-74X over ...

- 10 Designing computer systems with MEMS-based storage
Steven W. Schlosser, John Linwood Griffin, David F. Nagle, Gregory R. Ganger
November 2000 **ACM SIGPLAN Notices**, Volume 35 Issue 11

Full text available:  pdf(439.06 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

For decades the RAM-to-disk memory hierarchy gap has plagued computer architects. An exciting new storage technology based on microelectromechanical systems (MEMS) is poised to fill a large portion of this performance gap, significantly reduce system power consumption, and enable many new applications. This paper explores the system-level implications of integrating MEMS-based storage into the memory hierarchy. Results show that standalone MEMS-based storage reduces I/O stall times by 4--74X over ...

- 11 File and storage systems: The Google file system
Sanjay Ghemawat, Howard Gobioff, Shun-Tak Leung
October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**

Full text available:  pdf(275.54 KB) Additional Information: [full citation](#), [references](#), [index terms](#)

Keywords: clustered storage, data storage, fault tolerance, scalability

- 12 Using Handheld Devices in Synchronous Collaborative Scenarios
Jörg Roth, Claus Unger
January 2001 **Personal and Ubiquitous Computing**, Volume 5 Issue 4

Full text available:  pdf(284.67 KB) Additional Information: [full citation](#), [abstract](#), [index terms](#)

In this paper we present a platform specially designed for groupware applications running on handheld devices. Common groupware platforms request desktop computers as underlying hardware platforms. The fundamentally different nature of handheld devices has a great impact on the platform, e.g. resource limitations have to be considered, the network is slow and unstable. Often, personal data are stored on handheld devices, thus mechanisms have to ensure privacy. These considerations led to the Qui ...

- 13 An end-to-end approach to globally scalable network storage
Micah Beck, Terry Moore, James S. Plank
August 2002 **ACM SIGCOMM Computer Communication Review , Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications**, Volume 32 Issue 4

Full text available:  pdf(286.82 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper discusses the application of end-to-end design principles, which are characteristic of the architecture of the Internet, to network storage. While putting storage into the network fabric may seem to contradict end-to-end arguments, we try to show not only that there is no contradiction, but also that adherence to such an approach is the key to achieving true scalability of shared network storage. After discussing end-to-end


arguments with respect to several properties of network storage ...

Keywords: IBP, asynchronous communications, end-to-end design, exNode, internet backplane protocol, logistical networking, network storage, scalability, store and forward network, wide area storage

14 Axon: network virtual storage design

James P. G. Sterbenz, Gurudatta M. Parulkar

April 1990 **ACM SIGCOMM Computer Communication Review**, Volume 20 Issue 2

Full text available:  [pdf\(1.16 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This paper describes the design of *network virtual storage* (NVS) in the Axon host communication architecture for distributed applications. The Axon project is investigating an integrated design of host architecture, operating systems, and communication protocols to allow applications to utilise the high bandwidth provided by the next generation of communication networks. NVS extends segmented paged virtual storage management and address translation mechanisms to include segments ...

15 System support for pervasive applications

Robert Grimm, Janet Davis, Eric Lemar, Adam Macbeth, Steven Swanson, Thomas Anderson, Brian Bershad, Gaetano Borriello, Steven Gribble, David Wetherall

November 2004 **ACM Transactions on Computer Systems (TOCS)**, Volume 22 Issue 4

Full text available:  [pdf\(1.82 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Pervasive computing provides an attractive vision for the future of computing. Computational power will be available everywhere. Mobile and stationary devices will dynamically connect and coordinate to seamlessly help people in accomplishing their tasks. For this vision to become a reality, developers must build applications that constantly adapt to a highly dynamic computing environment. To make the developers' task feasible, we present a system architecture for pervasive computing, called & ...

Keywords: Asynchronous events, checkpointing, discovery, logic/operation pattern, migration, one.world, pervasive computing, structured I/O, tuples, ubiquitous computing

16 The Zebra striped network file system

John H. Hartman, John K. Ousterhout

August 1995 **ACM Transactions on Computer Systems (TOCS)**, Volume 13 Issue 3

Full text available:  [pdf\(2.76 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)


Zebra is a network file system that increases throughput by striping the file data across multiple servers. Rather than striping each file separately, Zebra forms all the new data from each client into a single stream, which it then stripes using an approach similar to a log-structured file system. This provides high performance for writes of small files as well as for reads and writes of large files. Zebra also writes parity information in each stripe in the style of RAID disk arrays; this ...

Keywords: RAID, log-based striping, log-structured file system, parity computation

17 The Zebra striped network file system

John H. Hartman, John K. Ousterhout

December 1993 **ACM SIGOPS Operating Systems Review , Proceedings of the fourteenth ACM symposium on Operating systems principles**, Volume 27 Issue 5

Full text available:  [pdf\(1.93 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Zebra is a network file system that increases throughput by striping file data across multiple servers. Rather than striping each file separately, Zebra forms all the new data from each client into a single stream, which it then stripes using an approach similar to a log-structured file system. This provides high performance for writes of small files as well as

for reads and writes of large files. Zebra also writes parity information in each stripe in the style of RAID disk arrays; this increase ...

18 Performing remote operations efficiently on a local computer network

Alfred Z. Spector

April 1982 **Communications of the ACM**, Volume 25 Issue 4

Full text available:  [pdf\(1.58 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


A communication model is described that can serve as a basis for a highly efficient communication subsystem for local networks. The model contains a taxonomy of communication instructions that can be implemented efficiently and can be a good basis for interprocessor communication. These communication instructions, called remote references, cause an operation to be performed by a remote process and, optionally, cause a value to be returned. This paper also presents implementation considerations ...

Keywords: communication models, efficient communication, transactions

19 Serverless network file systems

Thomas E. Anderson, Michael D. Dahlin, Jeanna M. Neefe, David A. Patterson, Drew S. Roselli, Randolph Y. Wang

February 1996 **ACM Transactions on Computer Systems (TOCS)**, Volume 14 Issue 1

Full text available:  [pdf\(2.69 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


We propose a new paradigm for network file system design: serverless network file systems. While traditional network file systems rely on a central server machine, a serverless system utilizes workstations cooperating as peers to provide all file system services. Any machine in the system can store, cache, or control any block of data. Our approach uses this location independence, in combination with fast local area networks, to provide better performance and scalability than ...

Keywords: RAID, log cleaning, log structured, log-based striping, logging, redundant data storage, scalable performance

20 Petal: distributed virtual disks

Edward K. Lee, Chandramohan A. Thekkath

September 1996 **Proceedings of the seventh international conference on Architectural support for programming languages and operating systems**, Volume 31, 30 Issue 9, 5

Full text available:  [pdf\(1.10 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The ideal storage system is globally accessible, always available, provides unlimited performance and capacity for a large number of clients, and requires no management. This paper describes the design, implementation, and performance of Petal, a system that attempts to approximate this ideal in practice through a novel combination of features. Petal consists of a collection of network-connected servers that cooperatively manage a pool of physical disks. To a Petal client, this collection appears ...

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)

 [QuickTime](#)

 [Windows Media Player](#)

 [Real Player](#)